IN THE CLAIMS:

The text of all pending claims, (including withdrawn claims) is set forth below. Cancelled and not entered claims are indicated with claim number and status only. The claims as listed below show added text with underlining and deleted text with ~~strikethrough~~. The status of each claim is indicated with one of (original), (currently amended), (canceled), (withdrawn), (new), (previously presented), or (not entered).

Please CANCEL claims 1 and 9, AMEND claims 2, 4, 5, 7, 8, 10, 12, 13, 15, 16, 17, 18 and 19 and ADD new claims 20, 21 and 22 in accordance with the following:

1. (CANCELED).

2. (CURRENTLY AMENDED) The computer as claimed in claim ~~1~~ 21, wherein said data holding part includes a plurality of registers.

3. (ORIGINAL) The computer as claimed in claim 2, said computer further comprising flags each of said flags indicating whether said data is held in said register.

4. (CURRENTLY AMENDED) The computer as claimed in claim ~~1~~ 21, said computer further comprising a data storing part, wherein said data holding part holds said data to be stored in said data storing part at a time when said interrupt occurs while a store instruction is executed, said store instruction requesting that said data is stored in said data storing part.

5. (CURRENTLY AMENDED) The computer as claimed in claim ~~1~~ 21, wherein said data holding part holds an instruction address of an instruction which causes said interrupt.

6. (CANCELED).

7. (CURRENTLY AMENDED) The computer as claimed in claim ~~1~~ 21, wherein said data holding part holds an effective address of a load instruction or a store instruction when said interrupt occurs while said load instruction or said store instruction is executed.

8. (CURRENTLY AMENDED) The computer as claimed in claim ~~1~~ 21, wherein said data is used for recovery from said interrupt.

9. (CANCELED).

10. (CURRENTLY AMENDED) The control method as claimed in claim ~~9~~ 22, wherein said data is held in a plurality of registers and said data is used for recovery from a plurality of interrupts.

11. (ORIGINAL) The control method as claimed in claim 10, wherein flags are used in which each of which flags indicates whether said data is held in said register.

12. (CURRENTLY AMENDED) The control method as claimed in claim ~~9~~ 22, said control method comprising the step of:

holding said data to be stored in a data storing part in said computer at a time when said interrupt occurs while a store instruction is executed, said store instruction requesting that said data is stored in said data storing part.

13. (CURRENTLY AMENDED) The control method as claimed in claim ~~9~~ 22, said control method comprising the step of:

holding an instruction address of an instruction which causes said interrupt.

14. (CANCELED).

15. (CURRENTLY AMENDED) The control method as claimed in claim ~~9~~ 22, said control method comprising the step of:

holding an effective address of a load instruction or a store instruction when said interrupt occurs while said load instruction or said store instruction is executed.

16. (CURRENTLY AMENDED) The control method as claimed in claim ~~9~~ 22, wherein said data is used for recovery from said interrupt.

17. (CURRENTLY AMENDED) A computer processing method, comprising:

holding in a memory at least an address of an instruction in an operation when interrupt processing that is not caused by the instruction causes the operation to halt; and

continuing the operation after the interrupt processing is discontinued.

18. (CURRENTLY AMENDED) The computer processing method according to claim 17, ~~further comprising~~ wherein said continuing of the operation ~~by executing~~ includes continuing execution of the instruction based on the address held in the memory, after the interrupt processing is discontinued.

19. (PREVIOUSLY PRESENTED) The computer processing method according to claim 17, wherein the address of the instruction is held in the memory when the interrupt processing starts to occur.

20. (NEW) The computer processing method according to claim 18, wherein the interrupt processing is initiated by an exception operation.

21. (NEW) A computer which processes an interrupt of a program caused by an exception operation when an instruction is executed, said computer comprising:

a data holding part holding data of said instruction that is interrupted by said interrupt at a time when said interrupt starts to occur; and

at least one instruction execution part using the data held by said data holding part to continue execution of said instruction without rerunning said instruction.

22. (NEW) A control method of a computer which processes an interrupt of a program caused by an exception operation while an instruction is executed, said method comprising:

holding data of said instruction that is interrupted by said interrupt at a time when said interrupt starts to occur; and

using the data held by said data holding part to continue execution of said instruction without rerunning said instruction.